

EXHIBIT F

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re U.S. Patent No: 7,777,651	R.C.N.: Not yet assigned
Issued: August 17, 2010	
Filed: June 2, 2008	Examiner: Not yet assigned
Applicants: Fallon, <i>et al.</i>	
Title: System and Method for Data Feed Acceleration and Encryption	Art Unit: Not yet assigned

REQUEST FOR *INTER PARTES* REEXAMINATION

Mail Stop *Inter Partes* Reexam
Central Reexamination Unit
Office of Patent Legal Administration
United States Patent & Trademark Office
P.O. BOX 1450
Alexandria, VA 22313-1450

TABLE OF CONTENTS

I.	INTRODUCTION.....	8
A.	Fees.....	8
B.	Claims Requested for Reexamination.....	8
C.	Certification per 37 C.F.R. § 1.907.....	9
D.	Terminal Disclaimer.....	9
E.	Related Applications.....	9
F.	Related Reexamination Proceedings.....	9
G.	Concurrent Litigation.....	9
H.	Service on the Patent Owner.....	10
I.	Disclaimer Regarding Claim Construction.....	10
II.	OVERVIEW OF THE PROSECUTION HISTORY.....	11
A.	The Claims of the '651 Patent Cannot Claim Priority to the 2000 Provisional Application or '987 Application.....	11
1.	The Data Field Specific Compression Claims (Claims 1, 52, 60, and 115) are at Best Entitled to the Priority Date for the '305 CIP Application filed May 7, 2003.....	12
2.	The Content Independent and Content Dependent Claims (Claims 22, 29, 73, and 98) are Only Entitled at Best to the Priority Date for the 2002 Provisional Application filed May 7, 2002.....	16
3.	The Descriptor-Based Fixed and Adaptive Table Claims for Data Blocks (Claims 13, 43, 91, and 108) Are at Best Entitled to the Priority Date for the '987 Application filed Oct. 3, 2001.....	21
B.	Prosecution History of the '651 Patent.....	24
C.	Issued Claims of the '651 Patent.....	26
1.	Broadest Reasonable Interpretation.....	26
2.	The Reasons for Allowance for the Claims of the '568 Patent and the 7,714,747 Patent are Applicable to the Claims of the '651 Patent.....	27
III.	IDENTIFICATION OF THE PRIOR ART REFERENCES THAT PRESENT SUBSTANTIAL NEW QUESTIONS OF PATENTABILITY	34
IV.	IDENTIFICATION OF THE SNQS AND CORRESPONDING PROPOSED REJECTIONS.....	36

V.	STATEMENTS POINTING OUT EACH SUBSTANTIAL NEW QUESTION OF PATENTABILITY PURSUANT TO 37 C.F.R. § 1.915(b)(3)	43
A.	Missing Subject Matter	43
	1. Missing Subject Matter for the Claims of the '651 Patent.....	44
B.	XMill Discloses a System for Compressing and Decompressing XML-Formatted Data	48
	1. Overview of XML-Formatted Data and XMill Compression.....	48
	2. XMill Discloses the Feature that Allegedly Distinguished the '651 Patent Claims Over the Prior Art Considered During Initial Examination.....	49
C.	Danskin Discloses a System for Compressing and Decompressing Structured Data.....	50
	1. Overview of the X Graphics Protocol and Structured Data	50
	2. Danskin Discloses the Feature that Allegedly Distinguished the '651 Patent Claims Over the Prior Art Considered During Initial Examination.....	54
D.	Sebastian Discloses a Network Architecture Incorporating Format Specific Compression.....	55
	1. Sebastian Discloses the Feature that Allegedly Distinguished the '651 Patent Claims Over the Prior Art Considered During Initial Examination.....	56
E.	Franaszek Discloses an Adaptive Multiple Dictionary Data Compression and Decompression System	57
	1. Framaszek Discloses the Feature that Allegedly Distinguished the '651 Patent Claims Over the Prior Art Considered During Initial Examination	58
F.	Carr Discloses a Data Processing Compression and Decompression System Including Data Packets.....	59
	1. Carr Discloses the Features that Allegedly Distinguished the '651 Patent Claims Over the Prior Art Considered During Initial Examination.....	60
G.	Hauck Discloses a Compression Device Using Run Length and Statistical Encoding.	61
	1. Hauck Discloses the Feature that Allegedly Distinguished the '651 Patent Claims Over the Prior Art Considered During Initial Examination.....	62
H.	XML-Xpress Discloses a System for Format-Specific Compression ..	63

1.	XML-Xpress Discloses the Feature that Allegedly Distinguished the '651 Patent Claims Over the Prior Art Considered During Initial Examination	65
I.	Reynar Discloses a Table Based Compression and Decompression System	65
J.	Zusman Discloses a System for Compressing and Broadcasting a Financial Data Feed	67
K.	Birdwell Discloses a System for Broadcasting Compressed Data	68
L.	Giltner Discloses a System for Table Based System for Encoding and Decoding Data	69
M.	RFC1662 Discloses a System for Frame Synchronization	71
N.	Langdon Discloses a Global and Adaptive Local Compression System	72
O.	Bhandari Discloses an Encoding and Decoding System Using State Machines and Tables	73
VI.	DETAILED EXPLANATION OF THE PERTINENCY AND MANNER OF APPLYING THE PATENTS AND PRINTED PUBLICATIONS PURSUANT TO 37 C.F.R. § 1.915(B)	76
A.	SNQ 1-6: Proposed Rejections based on XMill as the Primary Reference	76
1.	Ground 1: Claims 1-6, 10-12, 22-25, 29-32, 34, 38-42, 60-63, 67-78, 80, 84-90, 98-102, 106-107 are unpatentable under 35 U.S.C. § 103(a) as obvious over Xmill in view of Zusman.....	78
2.	Ground 2: Claims 7-9, 13-21, 26-28, 35-37, 43-59, 64-66, 81-83, 91-97, 103-105, 108-123 are unpatentable under 35 U.S.C. § 103(a) as obvious over XMill in view of Zusman further in view of Reynar.....	156
3.	Ground 3: Claims 33 and 79 are unpatentable under 35 U.S.C. § 103(a) as obvious over XMill in view of Zusman further in view of Birdwell.....	283
4.	Ground 4: Claims 1-6, 10-12, 22-25, 29-32, 34, 38-42, 60-63, 67-78, 80, 84-90, 98-102, 106-107 are unpatentable under 35 U.S.C. § 103(a) as obvious over XMill in view of XML-Xpress further in view of Zusman.	286
5.	Ground 5: Claims 7-9, 13-21, 26-28, 35-37, 43-59, 64-66, 81-83, 91-97, 103-105, 108-123 are unpatentable under 35 U.S.C. § 103(a) as obvious over XMill in view of XML-Xpress further in view of Zusman and further in view of Reynar.	382

6. Ground 6: Claims 33 and 79 are unpatentable under 35 U.S.C. § 103(a) as obvious over XMill in view of XML-Xpress further in view of Zusman and further in view of Birdwell.	525
B. SNQ 7-9: Proposed Rejections based on Danskin as the Primary Reference	529
1. Ground 7: Claims 1-6, 10-12, 22-25, 29-32, 34, 38-42, 60-63, 67-78, 80, 84-90, 98-102, 106-107 are unpatentable under 35 U.S.C. § 103(a) as obvious over Danskin in view of Zusman.	530
2. Ground 8: Claims 7-9, 13-21, 26-28, 35-37, 43-59, 64-66, 81-83, 91-97, 103-105, 108-123 are unpatentable under 35 U.S.C. § 103(a) as obvious over Danskin in view of Zusman further in view of Reynar.	619
3. Ground 9: Claims 33 and 79 are unpatentable under 35 U.S.C. § 103(a) as obvious over Danskin in view of Zusman further in view of Birdwell.	758
C. SNQ 10-14: Proposed Rejections based on Sebastian as the Primary Reference	762
1. Ground 10: Claims 1-3, 6-8, 10, 12-14, 17-19, 21-23, 25-27, 29, 30, 34-36, 38-44, 47-49, 51, 52, 55-58, 60, 63-65, 67-70, 72-76, 80-82, 84-87, 89-92, 95-100, 102-104, 107-109, 112-115, 118-121, 123 are unpatentable under 35 U.S.C. § 103(a) as obvious by Sebastian in view of Zusman.	764
2. Ground 11: Claims 4-6, 9, 15, 16, 18, 24, 25, 28, 31, 32, 34, 37, 45-47, 53-55, 61-63, 66, 77, 78, 80, 83, 93-95, 101, 102, 105, 110-112, 116-118 are unpatentable under 35 U.S.C. § 103(a) as obvious by Sebastian in view of Zusman further in view of Bhandari.	1042
3. Ground 12: Claims 11, 20, 50, 59, 71, 88, 90, 106, 122 are unpatentable under 35 U.S.C. § 103(a) as obvious by Sebastian in view of Zusman further in view of Birdwell.	1107
4. Ground 13: Claims 11, 20, 50, 59, 71, 88, 106, 122 are unpatentable under 35 U.S.C. § 103(a) as obvious by Sebastian in view of Zusman further in view of XML-Xpress.	1116
5. Ground 14: Claims 33, 79 are unpatentable under 35 U.S.C. § 103(a) as obvious by Sebastian in view of Zusman further in view of Bhandari further in view of Birdwell.	1126
D. SNQ 15: Proposed Rejections based on Franaszek as the Primary Reference	1129
1. Ground 15: Claims 1-123 are unpatentable under 35 U.S.C. § 103(a) as obvious over Franaszek in view of Zusman and further in view of Birdwell.	1130

E.	SNQ 16-19: Proposed Rejections based on Carr as the Primary Reference	1373
1.	Ground 16: Claims 1-3, 6-8, 10, 12-14, 17-19, 21-23, 25-27, 29, 30, 34-36, 38-44, 47-49, 51, 52, 55-58, 60, 63-65, 67-70, 72-76, 80-82, 84-87, 89-92, 95-100, 102-104, 107-109, 112-115, 118-121, 123 are unpatentable under 35 U.S.C. § 103(a) as obvious over Carr in view of Zusman	1374
2.	Ground 17: Claims 4, 5, 16, 24, 31-33, 46, 54, 61, 62, 77-79, 94, 101, 111, and 117 are obvious under 35 U.S.C. § 103(a) over Carr in view of Zusman further in view of RFC1662.....	1574
3.	Ground 18: Claims 9, 15, 28, 37, 45, 53, 66, 83, 93, 105, 110, 116 are unpatentable under 35 U.S.C. § 103(a) as obvious by Carr in view of Zusman further in view of Bhandari.	1597
4.	Ground 19: Claims 11, 20, 50, 59, 71, 88, 106, and 122 are obvious under 35 U.S.C. § 103(a) over Carr in view of Zusman further in view of Birdwell.	1617
F.	SNQ 20-23: Proposed Rejections based on Hauck as the Primary Reference	1625
1.	Ground 20: Claims 1-10, 12, 29-32, 34-42, 60-70, 72-78, 80-87, 89, 90, 98-105, 107 are unpatentable under 35 U.S.C. § 103(a) as obvious over Hauck in view of Zusman.	1626
2.	Ground 21: Claims 11, 33, 71, 79, 88, 106 are unpatentable under 35 U.S.C. § 103(a) as obvious over Hauck in view of Zusman and further in view of Birdwell.....	1705
3.	Ground 22: Claims 13-19, 21-28, 43-49, 51-58, 91-97, 108-121, 123 are unpatentable under 35 U.S.C. § 103(a) as obvious over Hauck in view of Zusman and Giltner.....	1710
4.	Ground 23: Claims 20, 50, 59, 122 are unpatentable under 35 U.S.C. § 103(a) as obvious over Hauck in view of Zusman and further in view of Giltner further in view of Birdwell.....	1809
G.	SNQ 24-26: Proposed Rejections based on XML-Xpress as the Primary Reference.....	1812
1.	Ground 24: Claims 1-3, 6, 10-12, 22, 23, 25, 29, 30, 34, 39-42, 60, 63, 68-75, 80, 85-90, 98-100, 102, 106, 107 are unpatentable under 35 U.S.C. § 103(a) as obvious by XML-Xpress in view of Zusman.....	1813
2.	Ground 25 Claims 4, 5, 24, 31, 32, 61, 62, 77, 78, 101 are obvious under 35 U.S.C. § 103(a) over XML-Xpress in view of Zusman further in view of RFC1662.	1886

3.	Ground 26: Claims 7-9, 26-28, 35-37, 64-66, 81-83, 103-105 are obvious under 35 U.S.C. § 103(a) over XML-Xpress in view of Zusman further in view of Langdon.	1898
H.	SNQ 27-29: Proposed Rejections based on Reynar as the Primary Reference	1920
1.	Ground 27: Claims 1-3, 6-15, 17-23, 25-30, 34-37, 39-45, 47-53, 55-60, 63-66, 68-75, 80-83, 85-93, 95-100, 102-110, 112-116, 118-123 are unpatentable under 35 U.S.C. § 103(a) as obvious over Reynar in view of Zusman.	1921
2.	Ground 28: Claims 4-5, 16, 24, 31-32, 46, 54, 61-62, 77-78, 94, 101, 111, 117 are unpatentable under 35 U.S.C. § 103(a) as obvious over Reynar in view of Zusman further in view of RFC1662.	2107
3.	Ground 29: Claims 33 and 79 are unpatentable under 35 U.S.C. § 103(a) as obvious over Reynar in view of Zusman further in view of RFC1662 and further in view of Birdwell.	2132
VII.	CONCLUSION	2136

Sebastian describes these filters as being included in a description file (called library) that indicates which lossless decoder (called filter) to use for the data field type:

The base includes the system control modules and a library used by all the filters.

(Sebastian, 1:52-53.)

In the Microsoft Windows environment, each Filter is a separate dynamically linked library (DLL) file. In addition to data tables, a filter includes executable code.

If a filter is found for a file, then it is dynamically linked and then called to encode the file. The encoded data stream includes an identification (ID) code indicating which filter was used to encode the data, and the decoder has to have a corresponding decoding filter.

To implement the filter selection system 22, the base module maintains a registry of the currently installed files, which includes the data need to identify source files of that type as well as the path to the DLL. This registry is updated each time a filter is added or replaced. The identification data can include both a list of file title suffices and a method to identify files via byte values near the beginning of the file. The identification procedure uses a search tree method to reduce the time needed to check the list when many filters are installed. From this registry, the appropriate filter 10 is invoked for the file type.

(Sebastian, 5:11-30) (emphasis added).

As to the fourth method step, Sebastian discloses decoding the data field with a selected lossless decoder utilizing content dependent data decompression, if the descriptor indicates the data field is encoded utilizing content dependent data compression. For example, Sebastian discloses decompressing the data field with a selected lossless decoder utilizing content dependent data decompression, if the descriptor indicates the data field is encoded utilizing content dependent data compression:

For each block, a compression algorithm is selected from a plurality of candidate compression algorithms and applied to the block. The compression algorithms can be determined based on the amount of data in the respective block. Furthermore, the compression algorithm can be adapted to the respective block, including the use of a customized transform. The selection of an algorithm can also be based on a compression model, which is derived from the format of the source data. The compressed data from the plurality of blocks are then combined into encoded data.

(Sebastian, 2:33-43.) As noted above, the encoded data includes a descriptor (called an identification code) identifying the data encoding used for the data. (*See*, Sebastian, 5:14-18.) According to Sebastian, “[f]or each block, a compression algorithm is selected from plurality of candidate compression algorithms and applied to the block.” (Sebastian, 33-35.) The decoder reverses this process. (*See* Sebastian, 3:36-42.) Sebastian further provides that:

each data component structured into a data field of an associated data type . . .

creating a plurality of blocks based on the source data structure, each block associated with a respective data field;

. . .

for each block:

selecting a compression algorithm from a plurality of candidate compression algorithms based on the data type of the data field associated with the block;

(Sebastian, claim 1, 27:43-61.) Sebastian further discloses:

This approach integrates the advantages of format-specific compression into general-purpose compression tool serving a wide range of data formats. The system includes filters which each support a specific data format, such as for Excel XLS worksheets or Word DOC files.

(Sebastian, 1:47-52.)

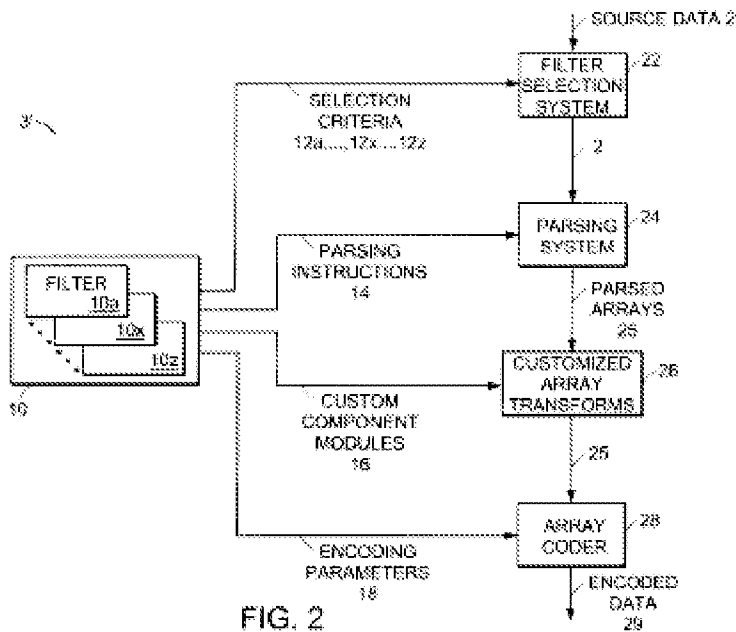
In particular at step 32, the Array 25 type is checked as a string array. If the Array is a string array, then a string coder 40 is called. If the array is not a string array, the Array type is checked at step 34 as a float type. If the Array is a float type, then a float coder 50 is called. If the Array is neither a string nor a float type, then it is an integer type and an integer coder 70 is called.

(Sebastian, 18:34-40.)

In addition, the system handles embedded data types with format-specific algorithms. “To handle compound file formats such as Microsoft’s OLE2, one filter can invoke another filter to encode a sub-file or stream within the compound file. For example, a Winword Filter might call an Excel Filter to encode some Excel data which was embedded within a compound document file.” (Sebastian, 5:31-36.)

As illustrated by Figure 2 below, if a filter is installed specific to a particular data type, then the system employs that filter. “The filter selection system 22 receives the source data

2 and checks the selection criteria 12a, . . . , 12x, . . . , 12z of all filters 10a, . . . , 10x, . . . , 10z installed in the system to see if any of them support the source data's format." (Sebastian, 4:9-12.) If there is no content-specific filter installed, "a 'generic' filter is used which provides compression performance similar to other generic compression systems, such as Lempel-Ziv (LZ) engines." (Sebastian, 4:12-15.) Sebastian thus provides a generic compression system for content-independent compression. (See Sebastian, 1:55-60; 4:9-20.) Figure 2 is illustrative of the disclosure in Sebastian:



With respect to the decoder, Sebastian provides:

The output of the encoder is encoded data 4, which can be stored in memory (in which case the encoding allows more source data to be stored within the same hardware), or transmitted to the decoder 5 (in which case the encoding the source data to be transmitted faster when the transmission channel bandwidth is limited). The decoder 5 retrieves or receives the encoded data 4 and reverses the algorithms used to encode the data so as to reconstruct the source data as decoded data 6. In a lossless system, the decoded data 6 is identical to the source data.

(Sebastian, 3:31-40.)

The descriptions to follow will mostly cover the encoder. The decoder just reverses the encoder's actions, so a description of the encoder is sufficient to describe the overall encoding/decoding system."

(Sebastian, 3:56-59.)

In one embodiment, Sebastian discloses compressing the array data as follows:

Array data is next tested at step 79 to determine if a matching algorithm should be applied to the data. The matching algorithms at step 81 eliminate redundant data items: those which completely match a previous entry in the array.

...

These Arrays are then converted, at step 87, to int32s so that the remaining routines can operate on a common data type.

(Sebastian, 20:30-46.)

As to the fifth method step, Sebastian discloses decoding the data field with a selected lossless decoder utilizing content independent data decompression, if the descriptor indicates the data field is encoded utilizing content independent data compression. For example, Sebastian discloses decoding the data field with a lossless decoder utilizing content independent data compression if the descriptor indicates that encoding utilized content independent data compression:

If a filter is installed which matches the format of the data to be encoded, the advantages of format-specific compression can be realized for that data. *Otherwise, a “generic” filter is used which achieves performance similar to other non-specific data compression systems (such as PKZip, Stacker, etc.).*

(Sebastian, 1:55-60 (emphasis added).)

The encoded data stream includes an identification (ID) code indicating which filter was used to encode the data, and the decoder has to have a corresponding decoding filter.

(Sebastian, 5:14-18.) Additional description of the content-independent compression is provided by Sebastian in the following disclosure:

The filter selection system 22 receives the source data 2 and checks the selection criteria 12a, ... , 12x, ... , 12z of all filters 10a, ... , 10x, ... , 10z installed in the system to see if any of them support the source data's format. *If not, a “generic” filter is used which provides compression performance similar to other generic compression systems, such as Lempel-Ziv (LZ) engines. In a particular preferred embodiment of the invention, the generic compression system employs an SZIP engine* as described by Mr. Schindler in U.S. application Ser. No. 08/970,220 filed

Nov. 14, 1997, the teachings of which are incorporated herein by reference in their entirety.

(Sebastian, 4:9-20) (emphasis added).

Customized array transforms 26 can usually improve the compression of these frequent components, while the automated default mechanisms achieve adequate compression of the remaining components.

(Sebastian, 16:45-49.) Sebastian describes decoding using a lossless decoder:

The decoder 5 retrieves or receives the encoded data 4 and reverses the algorithms used to encode the data so as to reconstruct the source data as decoded data 6. In a lossless system, the decoded data 6 is identical to the source data.

(Sebastian, 3:37-40.)

The descriptions to follow will mostly cover the encoder. The decoder just reverses the encoder's actions, so a description of the encoder is sufficient to describe the overall encoding/decoding system.

(Sebastian, 3:56-59.)

The un-matched values are then be [sic] converted to base 10 notation in a base 10 converter 53, as many databases include floating point values which were originally entered as base 10 values, and encoding the values through the array coder 28 in their native format is more efficient. The base 10 mantissas, exponents and conversion errors are processed separately by the array coder 28. For lossless coding an offset may have to be stored to correct slight rounding errors in the conversion.

(Sebastian, 19:1-10.) Sebastian also discloses “parsing each data component . . . based on the data field.” (Sebastian, Claim 1.)

Thus, as set forth in detail above, Sebastian in view of Zusman teaches each and every feature of independent claim 1. Accordingly, claim 1 is unpatentable under 35 U.S.C. § 103(a) as obvious by Sebastian in view of Zusman.

b) Claim 2

Claim 2 is dependent on claim 1, the limitations of which are disclosed as explained above. **As to the limitation added by claim 2, Sebastian discloses wherein the one or more lossless decoders are further selected based upon the specific ordering of the data**